

Component Documentation

A simple and effective visual separator for organizing content sections. Features multiple styling variants, horizontal and vertical orientations, and customizable colors for clear content separation.

How to use

Basic Usage

Simple divider implementations for content separation.

```
<aava-dividers  
  variant="solid"  
  orientation="horizontal"  
  color="#BBBEC5"  
></aava-dividers>
```

Variants

Different divider styles for various visual contexts.

```
<aava-dividers  
  variant="solid"  
  orientation="horizontal"  
  color="#BBBEC5"  
></aava-dividers>
```

```
<aava-dividers  
  variant="dashed"  
  orientation="horizontal"  
  color="#BBBEC5"  
></aava-dividers>
```

```
<aava-dividers  
  variant="dotted"  
  orientation="horizontal"  
  color="#BBBEC5"  
></aava-dividers>
```

```
<aava-dividers variant="gradient" orientation="horizontal"></aava-dividers>
```

Variant Types

- Solid : Clean, continuous line (default)
- Dashed : Dashed line pattern
- Dotted : Dotted line pattern
- Gradient : Smooth gradient effect

Orientation

Horizontal and vertical divider orientations for different layout needs.

```
<aava-dividers
  variant="solid"
  orientation="vertical"
  color="#dc3545"
></aava-dividers>

<aava-dividers
  variant="dotted"
  orientation="vertical"
  color="#ffc107"
></aava-dividers>
</div>

<aava-dividers
  variant="solid"
  orientation="vertical"
  color="#dc3545"
></aava-dividers>

<aava-dividers
  variant="dotted"
  orientation="vertical"
  color="#ffc107"
></aava-dividers>
```

API Reference

Inputs

Property	Type	Default	Description
variant	'solid' 'dashed' 'dotted' 'gradient'	'solid'	Visual style of the divider
orientation	'horizontal' 'vertical'	'horizontal'	Direction of the divider
color	string	'#000000'	Color of the divider (hex, rgb, or CSS color name)

CSS Custom Properties

Property	Default	Description
--divider-color	Component color input	Dynamic color applied to borders
--divider-background-gradient	Theme-based	Gradient background for gradient variant

Best Practices

Design Guidelines

- Use dividers sparingly to avoid visual clutter
- Choose appropriate variants based on content hierarchy
- Ensure sufficient color contrast for accessibility
- Consider spacing around dividers for optimal visual separation

Layout Usage

- Use horizontal dividers between content sections
- Use vertical dividers in sidebar layouts or columns
- Apply consistent divider styles throughout the interface
- Match divider colors to your design system

Accessibility & Dev Guidelines

The Divider component is purely visual and should not convey crucial content structure on its own. However, it's accessible:

- Uses role="separator"
- Honors aria-orientation="horizontal | vertical"
- When used with text, ensure surrounding context is semantic
- Test selector: data-testid="play-divider"

Do's and Don'ts

Do	Don't
Use for layout separation in dense UIs	Don't use as a replacement for headings
Style via tokens for consistent spacing	Don't hardcode padding/margins
Add labels only when useful context exists	Don't overuse labeled dividers visually
Use vertical divider inside horizontal layout	Don't nest labeled dividers unnecessarily
Use inset variant to separate nested blocks	Don't rely solely on divider for structure